

UNITED STATES DISTRICT COURT
DISTRICT OF MASSACHUSETTS

SCANSOFT, INC.)
Plaintiff,)
v.) Civil Action No. 04-10353 PBS
VOICE SIGNAL TECHNOLOGIES, INC.,)
LAURENCE S. GILICK, ROBERT S.)
ROTH, JONATHAN P. YAMRON, and)
MANFRED G. GRABHERR)
Defendants.)

DECLARATION OF RICHARD GOLDHOR, PhD

I, Richard Goldhor, declare as follows.

1. I am a consultant in the speech processing software industry, in which I have worked for the past 28 years. I have designed and developed speech recognition software and other innovations in speech processing technology for many companies including Kurzweil Applied Intelligence, Inc., once a leader in speech recognition research and development. I am also the inventor or co-inventor of several patents in the field. I have authored several publications on the design and implementation of speech recognition and speech processing technologies. I am currently principal and consultant of Grapevine Software, a consulting, research and development firm specializing in software development and research.
2. ScanSoft, Inc. ("ScanSoft") has asked me to comment on the computer text files produced by Voice Signal Technologies, Inc. ("VST") in connection with the above-captioned lawsuit, and in particular, how the computer software works and what information the source code contains.

3. In forming my opinions contained in this Declaration, I have examined a Compact Disk ("CD") provided to me by attorneys for ScanSoft. I understand VST has represented to ScanSoft that the CD contains the source code for the user interface portion of VST's VSuite product.
4. As a general matter, there are several deficiencies with regard to the CD that are so significant that I cannot render any substantive opinion about the source code provided on the CD. These deficiencies are outlined below.
5. I have observed that the CD contains four folders, or subdirectories, labeled: (1)

~~REDACTED~~

The attached Table 1 lists some basic information about each of these files.

6. Table 1 also demonstrates that each of the files in the CD contain references to multiple "include files" Include files are also known as "header files" and conventionally have names that end in ".h." Such files generally contain additional lines of source code, which may substantially augment or alter the intellectual content of the source code. An "include statement" appearing in the source code indicates that the statement is to be replaced with the contents of the specified header file. The header file itself may contain any number of statements including complex source code that implements complete algorithms. The effect of the include statement in the source code is to insert the contents of the include file into the source file at the point in the source file where the include statement appears. The include statement has the effect of making the contents of the "include" file part of the source code. Table 2 shows the list of include files referenced by name but not provided in the CD.

7. Because the include files have not been provided, the source code is incomplete, and has no single unambiguous meaning. That is, I cannot determine what the intellectual content of the source code is, because I only have access to part of it. I have confirmed this incompleteness by attempting to compile the source code. Compiling source code is the necessary first step in converting human-readable code into executable code. Any complete module of source code that conforms to the rules of the programming language in which it has been written can be compiled. Therefore, attempting to compile source code is an objective and reliable way to identify incomplete or incorrectly-written code.
8. In order to compile source code, the expert must know the programming language in which the source code is written. My review of the CD provided by VST suggests that the source code is written in the "C" programming language. However, when I attempted to compile the source code with a standard C compiler, I observed multiple "fatal" compilation errors. My attempts to use a standard C compiler to compile the source code failed because as stated in paragraph 5 above, a number of "include" files are missing from the CD. These fatal error messages are not surprising, because when a standard C compiler is attempting to compile source code, if the compiler cannot locate certain files in the source code, it will generate a fatal error and abandon the attempt to build an object file from that source code.
9. In my experience, there are typically multiple versions of a software product. There are many reasons for the existence of multiple versions of a software product. Perhaps the most obvious is that over time software developers create more complex versions of a product from earlier simpler versions. Software vendors are also likely to create multiple versions of a product in order to vary the functionality and capabilities of the software that they can offer customers.

As a result, the intellectual property content of a product will generally vary from version to version and in some cases may change substantially even though the change in intellectual property content may not be readily apparent or detectable at all to the average user of the product.

10. It is a standard professional practice for software developers and vendors to maintain repositories or archives of material from which they can regenerate any version of a product. These repositories are created and maintained by computer applications called version control systems. An important goal of any version control system is to allow a software developer to reliably recreate any important version of a software product *ab initio*. In essence, the version control system provides a history of the source code's development over time.
11. A standard version control system allows for every version of each file that was stored in the system to be faithfully maintained by the system along with detailed information about the times and dates when that file was modified. Thus, the contents of a version control repository serves as a useful and objective history of the development of a software product. The ability of a software vendor to rebuild versions at will is so valuable that considerable effort and resources are often devoted to supporting this activity. A corporate asset of roughly comparable value and commanding a comparable investment of resources might be a company's general accounting system for example.
12. The CD I examined does not contain a version control repository or archive. Given the absence of any version control information on the CD I examined, I am not able to determine with any reasonable degree of certainty at what point in time any modifications were made to the source code. Table 1 shows that the date and time stamps of the files on the CD do not represent plausible dates for the actual creation of their content. Fourteen of the files have a nominal creation

date of March 9, 2005, and a "last edited" date ranging from February 10, 2005 to March 8, 2005. The other two files have a creation date of January 23, 2104, and a blank "last edited" date.

13. In my opinion, additional computer files must accompany the source code in order for a technical expert to reach a reliable opinion about the intellectual content of the source code. Moreover, I am not able to determine with any reasonable degree of certainty (i) the relationship between the source code provided and the VSuite product; (ii) the substantive characteristics of the environment in which the source code was designed to be built; and (iii) how to successfully compile the source code in the programming language in which it was designed.
14. In my opinion, the requirements for developing reliable opinions on the intellectual content of VST's source code include: (1) information that identifies the particular version or release of the each of VST's speech recognition software products; (2) a specification of the target platform on which each product release is intended to operate (e.g. IBM, Windows, Macintosh); (3) information about the build and release tools used to create the product; (4) the specification and control files used with those tools to create the product; and (5) all source code components and other primary files necessary for rebuilding each relevant release component from the source code. These files are typically available to software vendors and programmers as part of the vendor's archival and repository system.

I DECLARE UNDER PENALTY OF PERJURY THAT THE FOREGOING IS TRUE AND
CORRECT. EXECUTED ON JULY 18, 2005.



RICHARD GOLDHOR, PhD.

Table 1: Basic properties of Subject Material files.

Noles:

¹ Anneximate value: actual number of undeclared identifiers may be larger.

¹² These creation dates are illegal values—that is, dates that would never appear if the values were set to record the actual date that the file was created. The fact that the “last modified” date listings are blank indicate that their stored date values are also illegal.

REDACTED

ED

REDACTED

REDACTED

Table 2: List of missing include files: each of these files are specified in one or more "#include" statements in the Subject Material, but are not included in that Material.